Securing OSPF Using Digital Signatures and Neighbor Checking

Casey T. Deccio Mark Clement
Kent Seamons
Computer Science Department
Brigham Young University
Provo, UT 84602
casey@byu.edu, clement@cs.byu.edu, seamons@cs.byu.edu

Abstract

Network reliability is becoming increasingly important as critical systems become reliant on the Internet. One of the most fundamental yet important areas of network reliability is the existing routing mechanisms. Because of the vital importance of this infrastructure, it is important that it remain resistant to attacks. We present in this paper a neighbor checking mechanism that allows routers running the Open Shortest Path First Protocol (OSPF) to validate routing information and prevent invalid information from being propagated throughout the network. This scheme, when combined with existing digital signatures techniques, secures OSPF routers against insider and outsider attack.

1 Introduction

The world has become very dependent on technology to complete common and vital tasks. Much of what is accomplished on a day to day basis relies on the stability of the underlying network infrastructure. With such a large and complex network of machinery comprising the Internet, a careful review of existing mechanisms is necessary to avoid overlooking possible vulnerabilities or security threats which could cause severe failure and loss of functionality. A failure in just a small portion of a large network could result in catastrophic loss of service [1, 7].

One of the most fundamental yet important areas of network reliability is the routing mechanisms in existence. It has been said that "abuse of the routing mechanisms and protocols is probably the simplest protocol-based attack available" [3]. In November 2002 the Beth Israel Deaconess Hospital suffered a major network outage due to routing problems [2]. Fortunately, the staff was able to revert to paper forms to meet patient needs, but the outage lasted for several days and could have resulted in the loss of human life. A deliberate attack on the routing infrastructure could produce worse results than were experienced in this incident. Acts of cyber-terrorism could in fact threaten national security.

Possible attacks on routing infrastructure include manipulating valid router information and traffic hijacking to gain unauthorized access or to slow service. A malicious router that advertises low metrics (distance

costs) is certain to attract a portion of the network traffic through itself, thus gaining significant control within the network. Any attack could be detrimental to the network.

One of the protocols used for network routing deployed in internal networks is the Open Shortest Path First Protocol (OSPF) [4]. OSPF is a *link-state* protocol, so called because each participating node is responsible to describing the state (e.g., links to neighboring networks, routers, and hosts) of its local neighborhood to all other nodes in the network. It has been established that OSPF can effectively determine efficient paths for traffic in an internal network. However, there are many vulnerabilities associated with OSPF that if not addressed can lead to network failure.

We present in this paper a mechanism for router authentication and integrity protection in OSPF, in an effort to prevent attacks aimed at the network routing infrastructure. Our work uses a concept called *neighbor checking* and relies on previous work [5, 6] involving digital signatures in OSPF. In the next section we provide an overview of the OSPF protocol. We then describe related research that has been performed in this area. In the section following we discuss our method of router authentication and outline several of its strengths and weaknesses.

2 The OSPF Protocol

In order to identify and secure potential vulnerabilities in the OSPF routing protocol, an understanding of the details of OSPF is necessary. This section provides an overview of the protocol specification [4] as it pertains to our research.

OSPF is widely used in *Autonomous Systems* (ASs)—internal networks managed by an entity, such as a business or university. In this protocol, each router is responsible for maintaining an identical database which describes the topology of the AS, so it can determine the shortest routes for packets to take to arrive at any particular destination. This *link-state database* is composed of *Link State Advertisements* (LSAs) received from other nodes in the AS, each containing information about the local neighborhood of the source node. To determine the shortest path to a destination, all routers run the exact same algorithm in parallel. Each uses its link-state database to construct a tree of shortest paths to all other destinations, with itself as the root. If there exist several equal-cost paths to a particular destination, then the traffic is distributed evenly among the routes.

2.1 Adjacencies

To build a description of the network topology each router must first establish *adjacencies* with its immediate neighbors. Routers first discover their neighbors by sending and receiving *Hello packets*. Using the *Hello protocol* Hello packets are sent out periodically on each network interface by each router. Once neighboring routers have 'met' via the Hello Protocol, the routers undergo a Database Exchange Process to enforce database synchronization. This synchronization is repeated periodically to maintain data integrity.

2.2 Flooding

Once adjacencies have been established between neighboring nodes, a node organizes the information about its local neighborhood (i.e., the information about the nodes it is directly connected to) into a LSA to distribute to all other nodes. The LSA reaches all other nodes through reliable, intelligent *flooding*. This process is as follows: when a router receives an advertisement from a neighbor, it acknowledges receipt of the advertisement

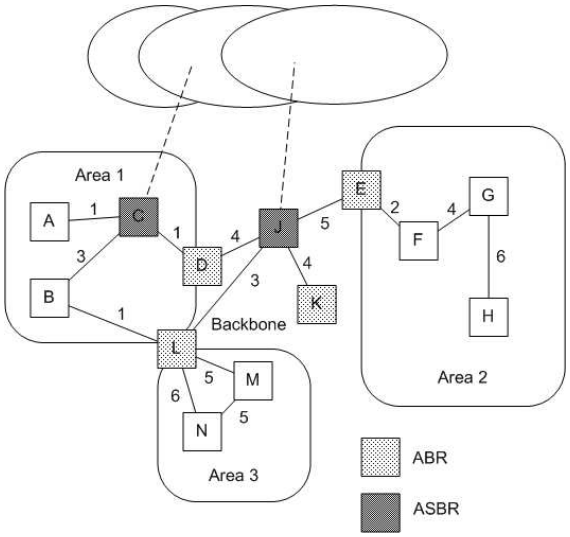

Figure 1: An example OSPF topology.

and, if the advertisement is new, forwards the advertisement to all other neighbors. Thus, all routers will have an identical topological database of LSAs from which they can derive shortest paths to all destinations.

2.3 Network Hierarchy

Routers can be grouped together in *areas* within an AS. Each area runs in parallel a copy of the basic link-state protocol. OSPF defines a two-level hierarchy among all areas in the AS: the top level is the *backbone*, and the next level consists of many areas connected to the backbone. A router which is part of two areas within the AS is known as an *Area Border Router* (ABR) and belongs to the backbone. Routers that are connected to points outside the AS are known as *Autonomous System Boundary Routers* (ASBRs). Routers within an area can determine the shortest path to a router outside the area using information from the ABRs. Likewise, routers within the AS can determine the shortest path to routers outside the AS via information from the ASBRs.

2.4 LSAs

LSAs are the building block for link-state databases. Each LSA describes the local neighborhood of the router where it originated (i.e., the nodes that are directly connected to it, and the metrics to travel to each in a single hop). OSPF defines five types of LSAs:

- *Type 1:* Each router sends a LSA to all the routers in its area, advertising its adjacencies.
- *Type 2:* Each multi-access network chooses a designated router through which traffic will be directed, so as to reduce traffic on the network. The designated router uses Network Links Advertisements to advertise the list of routers connected to the network.
- *Type 3:* Each ABR advertises a Summary Link Advertisement to each of the areas which it is attached to, describing routes to networks outside that area.
- *Type 4:* Each ABR advertises a Summary Link Advertisement to each of the areas which it is attached to, describing routes to ASBRs outside that area.
- *Type 5:* Each ASBR advertises many External Link Advertisements, which describe routes to destinations outside the AS.

The Type 1 LSA data from router `F` in Figure 1 would carry the LSA data shown in the following table:

		From		
		F	*E*	*G*
To	*F*			
	E	2		
	G	4		

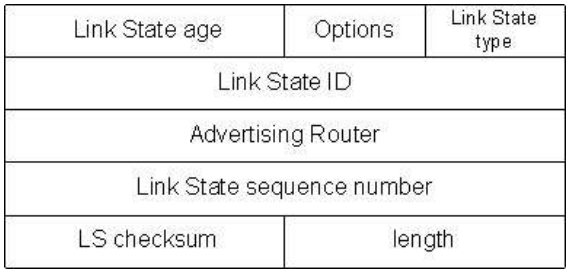

Figure 2: A LSA header.

In addition to carrying the state of its neighboring nodes, all LSAs carry a header that contains identifying information for that LSA:

- *Link State age:* The age of the LSA in seconds. At each router an LSA passes through the age is incremented by `InfTransDelay` until it reaches `MaxAge` (defined as one hour). LSAs with age `MaxAge` are not used in routing table calculation. Age is also incremented as the LSA resides in the database of any router. When the age of an LSA reaches `MaxAge` in a router's database, it is flushed from the database and re-flooded as a signal for other routers to remove it.

- *Options:* Indicates optional capabilities associated with the LSA.

- *Link State type:* The type of LSA (listed above).

- *Link State ID:* Identifies the piece of the routing domain being described by the LSA.

- *Advertising Router:* The ID of the OSPF router from which the LSA originated.

- *Link State sequence number:* A signed 32-bit integer used to identify each LSA. A router sends its first LSA with the lowest sequence number and sends each subsequent LSA with increasing incremental sequence numbers. If a router encounters more than one LSA from a particular router, the sequence number is used to determine which LSA is more recent. If the sequence numbers are equal, then the age, and finally the checksum are used as tie-breakers.

- *Link State checksum:* The checksum of the complete LSA excepting the Link State age field.

3 Related Work

Recent research in routing security has produced insightful analysis of the strengths and vulnerabilities of OSPF and other routing protocols. Several methods of router authentication and cryptography have been derived for securing these protocols against attackers.

3.1 Security Strengths in OSPF

In his research Wang [11] outlines some of the security strong points in the OSPF protocol. One of these advantages is found in the flooding process. If a faulty or compromised router is sending bogus information to a good router, a phenomenon known as *fight-back* ensues: the good router tries to *convince* the faulty router by repeatedly sending it good information.

Another advantage inherent in the link-state protocol is the property of *information least dependency*. LSAs contain raw information from the source router. In the event that an LSA containing incorrect data exists in the AS, this property allows other routers to identify the source of the problem. This differs from *distance-vector* protocols which designate that a router receive from its neighbors

an estimate of their cost to reach all other routers in the AS. The distance-vector routers then use this metric to calculate their own estimates to all other routers and then distribute them to their neighbors. Using this cost aggregation technique it is difficult for a router to validate the information it receives, and when bogus information is detected, it is difficult to determine its source.

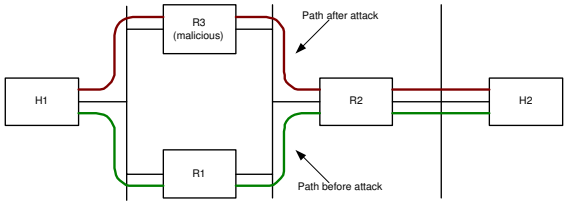

Figure 3: The paths of traffic between hosts $H1$ and $H2$ before and after a man-in-the-middle attack imposed by the malicious router $R3$.

3.2 Security Threats in OSPF

Wang [10] partitions attacks on routing infrastructure into two categories: *external* and *internal* attacks. We detail below these types of attacks and some preventative solutions.

3.2.1 External Attacks

External attacks are caused by non-protocol participants (*intruders*; e.g., an attacker with access to links between routers). Threats posed by outsiders include deletion, modification, replay, or forgery of protocol packets. These attacks could affect the adjacencies established in the AS and the update of routing tables of existing routers.

One specific external threat is the *man-in-the-middle* attack [11]. In this attack an intruder that has gained access to the network, posing as a valid router, advertises false metrics to hijack data traffic. Figure 3 displays such an attack, in which a malicious router, $R3$, has sent false metrics to its neighbors to intercept traffic. Once the malicious router has successfully redirected routes through itself, it can drop packets, manipulate data, or simply gain access to data that it is not authorized to view, without detection.

To guard against external attacks, OSPF Version 2 [4] requires that all protocol exchanges be *authenticated*, thus preventing an unauthorized router from posing as a valid router on the network. Every OSPF packet specifies an authentication type and designates a 64-bit field to be used for the authentication.

One possible authentication technique is the use of a 64-bit password [4]. However, any intruder with access to the network could trivially obtain a password sent in the clear using a *passive attack* (i.e., learning the password by observing network traffic) and thus authenticate itself.

Another authentication technique uses *cryptography* [4]. A shared secret key is configured for all routers within a subnet/network. For each OSPF packet, the key is used to generate/verify a *message digest* which is appended to the end of the packet. The digest is a one-way function of the secret key and the packet itself. The secret key is protected from passive attacks because it is never sent in the clear. Also, the increasing sequence number in the LSA prevents an attacker from replaying the packet and its corresponding digest.

3.2.2 Internal Attacks

Insider attacks occur when a router is faulty or has been compromised by an attacker. OSPF routers have the responsibility of generating local information (LSAs) and forwarding LSAs of other routers, so insider threats can attack either of these functions [10]. Possible attacks include generating bogus data and deletion or modification of cor-

rect routing information. Because insider attacks come from routers that appear to be recognized by the network, authentication schemes, such as those mentioned previously, will not protect the network against this type of attack; compromised routers have access to the passwords or secret keys.

One specific insider attack is the *premature aging* attack. In this attack the age field is manipulated by a participating router, to make it appear older than it is. When it prematurely reaches `MaxAge` in transit or in a particular router's link-state database, the LSA is flushed by the router and flooded to all other nodes. This will cause an incorrect description of the AS topology in link-state databases that have flushed the LSA.

3.3 Digital Signatures in OSPF

Some efforts have been made by Murphy to secure OSPF using *digital signatures* in a *public key infrastructure* (PKI) [5, 6]. In this method each *authenticated router* in an internal network maintains a public and private key. When an authenticated router sends routing protocol packets, it signs the data using its private key, thus preventing any participating router from manipulating the LSA in transit. All authenticated routers in the AS maintain a database with the public keys of all other authenticated routers. Using the public key of the LSA's owner, any router receiving the LSA can verify the LSA's integrity. Also, a router generating faulty information can be identified by its signature on the faulty OSPF packet.

3.3.1 Authenticated Routers and LSAs

Authenticated OSPF routers are capable of performing all the same functions as standard OSPF routers. In addition they generate signed routing information LSAs (*authenticated LSAs*), send *new key information LSAs*, manage key and signature algorithm verification, and verify signatures received. An authenticated LSA contains the following information:

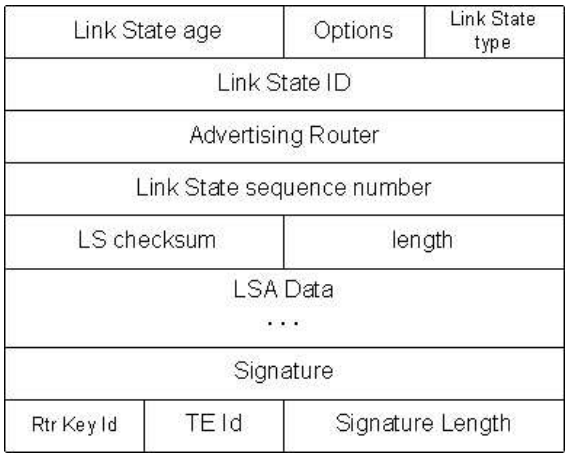

Figure 4: An authenticated LSA.

- *Normal LSA header:* The LSA header, as described in section 2.4.
- *Link state data:* Variable.
- *Signature:* The router's signature of the LSA, excluding the age.
- *Rtr Key Id:* Used to identify the router key used to sign this LSA.
- *TE Id:* The id of the Trusted Entity that produced the certificate.
- *Signature Length:* The length in bytes of the Signature.

The signature is produced by the router signing the LSA—all except the age. This allows authenticated routers through which the LSA passes to duly age it.

In order to guard against a *premature aging* attack, the signature of an authenticated LSA includes the age if and only if the age is maximum age. This prevents a router, other

than the LSA's owner, from flushing the LSA from the network.

3.3.2 Key Management and Distribution

Using the PKI described by Murphy [5, 6], a special router acts as a *Trusted Entity* for authenticated routers. This Trusted Entity maintains its own public and private key pair: its public key is known by all authenticated routers, and its private key is kept secret. In order for a router to be authenticated in this AS, it must flood the network with a *Router Public Key LSA*, which has the following components:

- *Normal LSA header:* The LSA header, as described in section 2.4.
- *Signature information:* Includes the information to correctly interpret the signature, as in the authenticated LSA above.
- *Certified information:* The information that the Trusted Entity has certified (which must be the same as in the LSA header).
- *Certification:* The signature produced by the Trusted Entity of the certified information.
- *Signature:* The router's signature of the LSA, excluding the age.

The Router Public Key LSA contains information that must be certified by the Trusted Entity: the router id; the public key; the router's role in the AS (e.g., internal router, ABR, ASBR); and the key's expiration time. The whole LSA (minus the age) is signed by the originating router, and this signature is also included. Any authenticated router receiving this Router Public Key LSA will add the new authenticated router's id and public key to its database. Because the addition of new routers to a network is generally *planned* by network administrators, the necessary configurations for a new router to join an authenticated network may be performed offline, thus avoiding the the risk of the Trusted Entity being compromised.

3.3.3 Advantages and Disadvantages

One of the advantages of the digital signature scheme proposed by Murphy is that the integrity of each LSA is maintained as the LSA traverses the network. This is an improvement over the shared secret key technique in the respect that LSAs are digitally bound to their owners, and helps prevent internal attacks. This also makes it easier to pinpoint problems in the network.

A large disadvantage of using a digital signature scheme is the computational complexity required to perform signature generation and verification. Network routers must minimize the delay involved in forwarding data packets. In order to achieve maximum performance, computations must be kept to a minimum.

Performance analysis of the digital signature security scheme shows that the greatest effects on performance are caused by the computation required for signature verification; an LSA is signed only once but verified by every router that receives it in the flooding process [5]. One suggestion that Murphy proposes to overcome this computational overhead is the implementation of a processor dedicated to the purpose of verifying the digital signatures of received routing packets. Wang [10] proposes another scheme using *selective verification* in which every LSA is signed, but the signatures are only verified by the other routers if the originator has detected modification. In this case the LSA's originator *enables* verification of its LSAs via

a flooded signal.

3.4 Predecessor Information

Although OSPF is a link-state protocol, we here summarize concepts that have been applied to secure distance-vector routing protocols because they pertain to this research of securing link-state protocols also. In researching methods for protecting distance-vector routing protocols, Smith [8,9] has proposed the use of predecessor information to verify integrity of routing data. Routing updates in distance-vector protocols contain multiple entries, each specifying a destination node and the metric to that destination. Using the predecessor method, the updates also contain information about the second-to-last hop (predecessor) to the destination. By including this information, the receiving router can verify the entire path from itself to the destination. This is accomplished by verifying the path first to the predecessor using information reported by the routers directly adjacent to the destination. This is iteratively repeated for each intermediate hop in the path.

4 Neighbor Checking

This research uses several concepts from the papers referenced herein to secure the OSPF protocol. We propose a security mechanism that includes digital signatures and neighbor checking—a concept related to predecessor information.

4.1 Motivation

An analysis of the digital signature approach to securing OSPF network produces solid results against outsider attacks. Insider attacks in which a router produces bogus LSAs can also be detected and traced to the guilty party. However, even the digital signature pattern leaves room for a router to manipulate an LSA *en route* and go undetected.

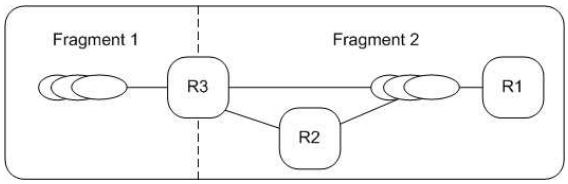

Figure 6: A partitioned network. Router $R3$ is the only link between the two fragments.

The digital signature authentication previously outlined [5, 6] guards against a router prematurely setting the age of a passing LSA to `MaxAge` because the LSA's signature includes the age field when and only when the LSA's age is `MaxAge`. However, there is nothing to prevent a router from prematurely aging a LSA to some older age less than `MaxAge`.

Figure 5 shows what happens when a router $R2$ prematurely ages an LSA before passing it on to $R3$. $R3$ ages the LSA to `MaxAge` and subsequently flushes it and then floods it to its neighbors. This may not seem to be a problem because the neighbors of $R3$ will not flood the LSA with an age field of `MaxAge` because it is not signed by the LSA—which would include the age field in its signature. However, the OSPF Version 2 specification [4] states that if only one of two LSAs has its age field set to `MaxAge`, then that LSA is the more recent of the two. This dictates that if $R3$ ever receives a "good" version of the LSA, $R3$ will discard it as outdated. To further the extent of the problem we consider a case in which $R3$ is the only link between two fragments of an OSPF area [10] (see Figure 6). If $R3$ received the aged packet in this state, then no LSAs from Fragment 2 will ever reach Fragment 1. This will keep Fragment 1 from updating critical information and may disable parts of the network.

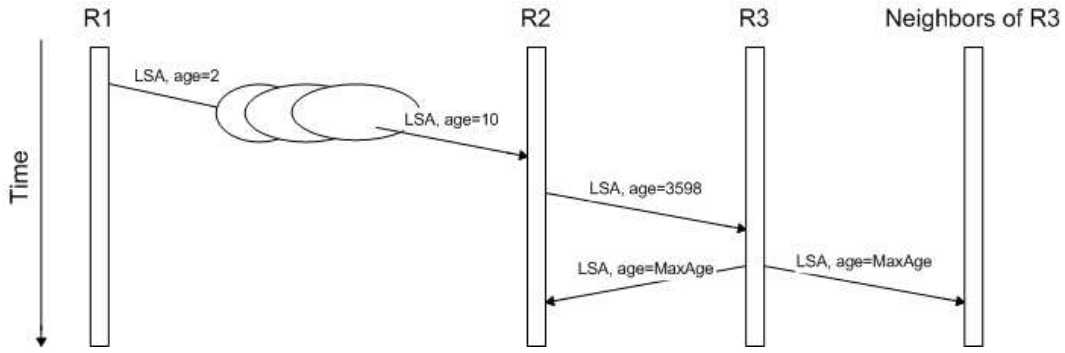

Figure 5: A premature aging attack in which $R2$ prematurely ages an LSA, following which $R3$ ages the LSA to MaxAge and floods it to its neighbors. Its neighbors do not accept the packet because it is not appropriately signed.

4.2 Methodology

The notion of neighbor checking stems from the fact that when an LSA is flooded from any given node, it must first pass through the first-generation neighbors of that particular node. By including in the LSA information from the previous hop, a router can verify the integrity of the LSA's data, all the way back to its source. This idea is similar to the predecessor method proposed by Smith [8, 9]. If a router can verify that information added by the neighbor from which it arrived is correct, then it can be shown recursively that every hop is correct back to the LSA's originator.

The algorithm is straightforward. When a router originates an LSA and floods it to the network, it gives the age field an initial value of 0. At the first hop from the LSA's origin, the receiving router recognizes that the previous hop was the original source of the LSA, so no verification is needed. It copies the data in the *current age* field to an identical field designated for predecessor information—the *previous age* field—and updates the current age field of the LSA by adding `InfTransDelay` to it. At every subsequent hop, the router receiving this LSA from neighbor and predecessor follows a similar pattern:

- Verify that the current age in the LSA is equal (within a predefined error) to the sum of the age in the predecessor field and `InfTransDelay`.
- If the current age is incorrect, then deny the packet and don't continue.
- Else, replace the predecessor age with the current age, update the current age field by adding `InfTransDelay` to it.
- Flood the LSA to neighbors.

In perspective, we examine the premature aging attack explained previously. When $R3$ in Figures 5 and 6 receives the LSA with the age of 3598, $R3$ analyzes the packet to test its validity. Because the packet also contains the age from the hop before $R2$, $R3$ can detect that the LSA has been prematurely aged by $R2$. $R3$ discards the LSA, and when a correct version of the LSA reaches $R3$ through a different router, it can finish flooding the network.

4.3 Implementation

The neighbor checking mechanism has all the functionality of the digital signature scheme outlined by Murphy [5, 6]. In addition it

contains two new fields and their accompanying signatures in the header: the *current age* field—signed by the neighbor from which the LSA came—and the *previous age* field—signed by the router that sent it to that neighbor. The signatures of these fields are contained in the header along with the signature information, so routers know how to interpret them. The signatures themselves are derived from the current age (or previous age) field and the LSA sequence number, to prevent against an inside router replaying the data.

4.4 Benefits and Drawbacks

One of the largest benefits of the neighbor checking algorithm is that it can isolate and identify problem routers before they cause network difficulties. Because the neighbors of problem routers will deny suspicious packets from them, the OSPF protocol will essentially work around them—that is, exclude problem routers altogether from the protocol. This also prevents bogus routing information from penetrating a circle of first-generation neighbors and propagating through the network.

Although using neighbor checking with signed age fields reduces the risk of attack, it adds to the computational overhead of routing. However, the neighbor checking concept applied in this research need not always include digital signatures; the current and previous age fields can enforce age integrity without signatures, and the feature of digital signing could be turned off except in cases of heightened paranoia. Future research could quantify the problem of overhead and identify potential solutions.

5 Conclusion

This research presents a mechanism for securing the OSPF protocol against outsider and insider attacks of several types. Using a neighbor checking scheme OSPF routers can verify integrity of the LSAs in transit. When coupled with a digital signature scheme, the OSPF protocol is less vulnerable to attack.

This research is necessary to secure the current routing infrastructure in order to protect users from malicious attacks and network failures. It improves security features in existing algorithms and can improve the reliability and reduce the vulnerability of critical OSPF routing systems. These improvements to OSPF can help avert catastrophic failures such as the one experienced at Beth Israel Hospital.

References

[1] Réka Albert, Hawoong Jeong, and Albert-László Barabási. The Internet's Achilles' Heel: Error and attack tolerance of complex networks. *Nature*, 406:378, 2000.

[2] Ann Bednarz. Hospital sounds alarm after 3-day struggle. *Network World*, Nov 2002.

[3] S. M. Bellovin. Security problems in the TCP/IP protocol suite. *Computer Communications Review*, 19(2):32–48, 1989.

[4] J. Moy. OSPF version 2. RFC 2328, Apr 1998.

[5] S. Murphy, M. Badger, and B. Wellington. OSPF with digital signatures. RFC 2154, Jun 1997.

[6] S. L. Murphy and M. R. Badger. Digital signature protection of the OSPF routing protocol. pages 93–102. The Symposium on Network and Distributed System Security, Feb 1996.

[7] G. Qu, J. Rudraraju, R. Modukuri, S. Hariri, and C. Raghavendra. A frame-

work for network vulnerability analysis. IASTED International Conference on Communications, Internet, and Information Technology (CIIT 2002), Nov 2002.

[8] B. Smith and J. Garcia-Luna-Aceves. Securing the border gateway routing protocol. London, UK, Nov 1996. Global Internet '96.

[9] Bradley R. Smith, Shree Murphy, and J. J. Garcia-Luna-Aceves. Securing distance-vector routing protocols. San Diego, CA, Feb 1997. Symposium on Network and Distributed System Security.

[10] B. Vetter, F. Wang, and S.F. Wu. An experimental study of insider attacks for OSPF routing protocol. pages 293–300. The 1997 International Conference on Network Protocols, Oct 1997.

[11] F. Wang and S. Wu. On the vulnerabilities and protection of OSPF routing protocol. pages 148–152. The 7th International Conference on Computer Communications and Networks, Oct 1998.